Kapitola: Primitívna rekurzia

1.1 Motivácia

Matematických funkcií je nespočítateľne veľa. Dokonca už unárnych funkcií na $\mathbb{N}$ je nespočítateľne veľa. (Ľahký dôkaz úpravou Cantorovej diagonalizácie.)

Tým pádom ale väčšina z nich pre nás nie je zaujímavá – nikdy nikde takéto funkcie nestretneme. Istá zaujímavosť začína až u funkcií, ktoré vieme exaktne definovať.

Samotná definícia však ešte nemusí byť všetko. Uvažujme napríklad funkciu:

$$fives(x) = \begin{cases} 1 & \leftarrow \text{v desatinnom rozvoji } \pi \text{ sa niekde nachádza } x \text{ pätiek za sebou} \\ 0 & \leftarrow \text{inak} \end{cases}$$

Toto je síce exaktná definícia, jasne popisujúca práve jednu funkciu – je nám ale nanič. Nehovorí nám v podstate nič o tom, ako túto funkciu vypočítať. Je dokonca otvorený problém či je $fives$ identicky rovná jednej.

Všimnite si tiež rozdiel medzi „(na základe definície) nevieme funkciu vyhodnotiť" a „neexistuje algoritmus ktorý by túto funkciu vyhodnocoval". To prvé je v tomto prípade (zatiaľ) pravda, to druhé však nie. Totiž určite existuje triviálny program, ktorý funkciu $fives$ ráta. My len nevieme, či je to program:

```
return 1;
```

alebo jeden z programov tvaru:

```
if (vstup < K) then return 1 else return 0;
```

Pekné funkcie by teda boli také, u ktorých je definícia priamo „návodom" na ich počítanie.

Ako takéto definície robiť systematicky? Pre úvod: Ako ich robiť systematicky pre funkcie na $\mathbb{N}$?

Pekná funkcia je napríklad $f(x, y) = x^y$. Ako to vyzerá, keď ju vyhodnocujeme? Hodnotu x^y dostaneme tak, že zoberieme 1ku a y-krát ju vynásobíme hodnotou x.

Čo znamená „vynásobíme hodnotu a hodnotou b"? Zoberieme nulu a b-krát k nej pripočítame a.

A čo znamená „pripočítame"?

Takto sa môžeme na zložité procesy vyhodnocovania funkcií dívať ako na veľa malých, jednoduchých krokov. Otázka znie, kde už sme nútení prestať, čo sa už nedá ďalej zjednodušiť?

Taktiež nás tento príklad môže inšpirovať k voľbe nástroja na veľmi jednoduchý popis mnohých funkcií: rekurziu. Napr. „x umocnené na y" môžeme definovať pomocou predpisu „x umocnené na $n+1$" je „x umocnené na n" krát x. (Plus samozrejme potrebujeme začiatočné podmienky, v tomto prípade: „x umocnené na 0" je 1.)

1.2 Definícia

Trieda $PREC$ primitívne rekurzívnych funkcií je definovaná nasledovne:

1. Nulárna funkcia $z \equiv 0$ je v $PREC$
2. Unárna funkcia $s(x) = x + 1$ je v $PREC$. Túto funkciu voláme nasledovník (successor).
3. Pre každé $1 \leq k \leq n$ je n-árna funkcia $P_k^n(x_1, \ldots, x_n) = x_k$ v $PREC$ Tieto funkcie voláme projekcie.
4. Ak $f \in PREC$ je k-árna funkcia a $g_1, \ldots, g_k \in PREC$ sú m-árne funkcie, potom m-árna funkcia $h(x_1, \ldots, x_m) = f(g_1(x_1, \ldots, x_m), \ldots, g_k(x_1, \ldots, x_m))$ je v $PREC$. Hovoríme, že h vzniká kompozíciou pôvodných funkcií.
5. Ak $f \in PREC$ je k-árna a $g \in PREC$ je $(k+2)$-árna funkcia. Potom $h \in PREC$, kde $h(0, x_1, \ldots, x_k) = f(x_1, \ldots, x_k)$ a $h(s(n), x_1, \ldots, x_k) = g(h(n, x_1, \ldots, x_k), n, x_1, \ldots, x_k)$. Hovoríme, že h vzniká z f a g primitívnou rekurziou.
6. V $PREC$ nie sú žiadne iné funkcie.

Ekvivalentne, $PREC$ je najmenšia trieda funkcií, ktorá obsahuje funkcie z bodov 1-3 a je uzavretá vzhľadom na kompozíciu a na primitívnu rekurziu.

Všimnite si, že operácia primitívnej rekurzie je „pekná“. Vyhodnotenie konkrétnej hodnoty funkcie h môže vyžadovať vyhodnotenie h aj pre iné hodnoty. Môžeme si však všimnúť, že hodnota prvého parametra odovzdávaného funkcii h pri každom kroku klesá, preto náš výpočet určite skončí po vyhodnotení h pre konečne veľa vstupov.

Všetky funkcie v $PREC$ sú totálne: dôkaz indukciou podľa definície.

1.3 Príklady

1.3.1 Konštanty

Konštanty, presnejšie nulárne funkcie, vieme vyrobiť postupnou kompozíciou veľa successorov a nuly.

Formálne môžeme napríklad nulárnu funkciu $j \equiv 1$ vyrobiť nasledovne: $j \equiv Comp[s, z]$. (Slovne: j vzniká kompozíciou z s a z.)

Unárna 0, teda funkcia $z_1(x) = 0$, vzniká primitívnou rekurziou zo z a P_1^2: podľa tejto definície je $z_1(0) = z = 0$ a $\forall n : z_1(n+1) = z_1(n)$.

Ostatné unárne konštanty dostaneme postupnou kompozíciou successorov a unárnej nuly.

Nulu s vyššou aritou dostaneme primitívnou rekurziou z unárnej nuly a P_1^n pre $n = 3, 4, \dots$ a ďalej to už je zjavné.

1.3.2 Identita

Identitu máme samozrejme v našej množine projekcií, je to $i \equiv P_1^1$.

1.3.3 Sčítanie

Intuitívne: sčítanie môžeme definovať nasledovne:

$$
\begin{aligned}
add(0, y) &= y \\
add(x+1, y) &= add(x, y) + 1
\end{aligned}
$$

Podľa tejto intuitívnej definície vyrobíme formálnu, pomocou operácie primitívnej rekurzie. Na jej použitie potrebujem mať dve funkcie. Funkcia f popisujúca základný prípad má v našom prípade pre ľubovoľný vstup y vrátiť výstup y, je to teda identita.

Funkcia g, ktorá počíta rekurzívny krok, dostane vstupy $add(y, x)$, y a x a má vrátiť výstup $add(y, x) + 1$. Teda ide o funkciu $g(u, v, w) = u + 1$. Takáto funkcia je zjavne primitívne rekurzívna, lebo vzniká kompozíciou s a P_1^3.

Formálne teda $add \equiv PR[P_1^1, Comp[s, P_1^3\ ]\ ]$.

1.3.4 Ďalej ľahko zostrojíme

Predecesor, odčítanie (ktoré nevie podtiecť pod 0), signum, absolútna hodnota rozdielu, násobenie, umocňovanie, štvorec, tretiu mocninu, ľubovoľný polynóm s prirodzenými koeficientami, faktoriál, n-árne minimum a maximum, predikát pre deliteľnosť, podiel, zvyšok, predikáty pre rovnosť a nerovnosti dvoch hodnôt, ...

1.3.5 Neat trick

Pomocou signu vieme ľahko dokázať „vetu o if-e“:

Nech $f_1, \dots, f_{k+1}$ a $g_1, \dots, g_k$ sú primitívne rekurzívne funkcie s rovnakou aritou m. Položme $g_{k+1} \equiv 1$ a definujme funkciu h nasledovne: $h(\overline{x})$ je rovné $f_i(\overline{x})$, kde i je najmenšie prirodzené číslo, pre ktoré $g_i(\overline{x}) > 0$. Potom h je tiež primitívne rekurzívna.

Myšlienka dôkazu: $sgn(g_1(\overline{x})) \cdot f_1(\overline{x})$ je buď $f_1(\overline{x})$ alebo 0, podľa toho, či $g_1(\overline{x}) > 0$.